ARI Research Note 98-05

The Virtual Schoolhouse

John Leddo, James Kolodziej, Zhixiong Zhang, and
Sean Beary
Research Development Corporation

Infantry Forces Research Unit
Scott E. Graham, Chief

March 1998

United States Army
Research Institute for the Behavioral and Social Sciences

Approved for public release; distribution is unlimited.

U.S. Army Research Institute for the Behavioral and Social Sciences

A Directorate of the U.S. Army Total Personnel Command

EDGAR M. JOHNSON
Director

Technical Review by

Robert J. Pleban

NOTICES

DISTRIBUTION: This Research Note has been cleared for release to the Defense Technical Information Center (DTIC) to comply with regulatory requirements. It has been given no primary distribution other than to DTIC and will be available only through DTIC or the National Technical Information Service (NTIS).

FINAL DISPOSITION: This Research Note may be destroyed when it is no longer needed. Please do not return it to the U.S. Army Research Institute for the Behavioral and Social Sciences.

NOTE: The views, opinions, and findings in this Research Note are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision unless so designated by other authorized documents.

REPORT DOCUMENTATION PAGE

1. REPORT DATE (dd-mm-yy) 03-98	2. REPORT TYPE FINAL	3. DATES COVERED (from. . . to) September 1996 through June 1997
4. TITLE AND SUBTITLE The Virtual Schoolhouse		5a. CONTRACT OR GRANT NUMBER DASW01-96-C-0069
		5b. PROGRAM ELEMENT NUMBER 876600
6. AUTHOR(S) John Leddo, James Kolodziej, Zhixiong Zhang, and Sean Beary		5c. PROJECT NUMBER M861
		5d. TASK NUMBER 6901
		5e. WORK UNIT NUMBER C14
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Research Development Corporation 2875 Towerview Road Suite A4 Herndon, VA 22071		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Research Institute for the Behavioral and Social Sciences 5001 Eisenhower Avenue Alexandria, VA 22333-5600		10. MONITOR ACRONYM ARI
		11. MONITOR REPORT NUMBER Research Note 98-05

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

Scott E. Graham, Contracting Officer Representative

14. ABSTRACT *(Maximum 200 words)*:

The report describes a Phase I Small Business Technology Transfer (STTR) project in which a Distributed Interactive Intelligent Tutoring Simulation ™ (DIITS) was developed to train Army Infantry squad and fire team leaders the skills they need to cooperatively perform military operations in urban terrain (MOUT). The intelligent tutoring system technology allowed trainees to receive feedback and remediation regardless of whether or not a human instructor was present. The DIITS included intelligent agent technology to play the role of scenario agents when a human was not available to fill in. This gave the technology added power as it could be used for training regardless of the number of trainees available at the time. A scenario editor was also created to allow training scenarios to be developed by users. The intention of this was to increase the customizability of the technology to individual user needs. Finally, the technology was constructed to be generic and modular to support extension and reuse as training requirements evolve. These characteristics were demonstrated in several ways including the transfer of technologies across projects, the substitutability of modules across systems and the ability of the technology to respond to user-defined scenarios without further modification.

15. SUBJECT TERMS

MOUT   Intelligent tutors   Urban fighting   Distributed training

SECURITY CLASSIFICATION OF			19. LIMITATION OF ABSTRACT	20. NUMBER OF PAGES	21. RESPONSIBLE PERSON (Name and Telephone Number)
16. REPORT	17. ABSTRACT	18. THIS PAGE			
Unclassified	Unclassified	Unclassified	Unlimited	29	

THE VIRTUAL SCHOOLHOUSE

CONTENTS

LIST OF TABLES

LIST OF FIGURES

THE VIRTUAL SCHOOLHOUSE

Introduction

Training is undergoing a substantial change in the military. This is a result of a variety of factors. First, budget cuts within the department of defense has both reduced the size of the fighting force as well as the dollars allocated to training that force. However, with weapons systems becoming increasingly complex and decreasing educational achievement of the student population from which the military draws its recruits (cf., National Assessment of Educational Progress, 1992), the military is forced to train more with a smaller budget. This is all the more exacerbated by the requirement that the military prepare for a wider range of missions, conflicts and theaters than ever before, all with a smaller fighting force. The level of preparedness that soldiers must attain to accomplish these missions is greater than ever.

These factors have forced changes in the way the military trains its personnel. The most visible change is the trend to move training from the traditional schoolhouse to the field (cf. Dyer, 1996). This accomplishes several of the objectives discussed above. First, there is a tremendous cost both in real dollars and opportunity to bring soldiers to the schoolhouse to receive training. In addition to travel and per diem costs ("real dollars"), soldiers spend time away from their units and hence temporarily suspend performance of their regular duties ("opportunity costs"). Given the downsizing of the military both in budget and size of the active force, both of these costs represent a strain on the goal of providing a ready, well-trained fighting force.

Second, there is tremendous concern that the effects of schoolhouse training may degrade before the skills that are trained are actually put to use on the job. This is particularly true of National Guard and Reserve personnel that see duty only a fraction of the year and have little opportunity to practice and maintain military (or MOS) skills while they are performing their "full-time" jobs. In fact, Leddo et al. (1990) found that most experts they studied felt that the bulk of their expertise was acquired through on the job training rather than at the schoolhouse. This type of finding has led to the notion of "just in time" training, where training is delivered to the job site when it is needed so that the skills being trained will be at their sharpest when they are actually used. The need for "just in time" training is at a high and will continue to grow.

Third, with a reduction in the fighting force of the Army, the pool of expertise is shrinking. Military experts are simply retiring and their expertise retires with them. As a result, the pool of instructors has also shrunk. This creates a need to leverage existing expertise.

The motivation to reduce training costs, to make training more timely and robust, and to maximize the use of diminishing expertise is driving current Army thinking of how to push training from the schoolhouse to the field.

Current Approaches to Moving Training to the Units

Technology is becoming an integral part of this transfer of training to the field. Computers and telecommunications technologies offer the opportunity to disseminate training to a world-wide Army. Further, this dissemination can occur virtually instantly allowing soldier to receive the "latest and the greatest". By moving the training from the schoolhouse to the field, it comes under local control and can be made more responsive to individual unit training needs. Finally, once under field control, soldiers can receive training on the job, which saves additional costs of instructor time and travel to bring the soldier to a schoolhouse.

In considering how technology may play a role in this shift, we consider the benefits a schoolhouse provides that may be desirable to preserve as the training moves to the unit. A physical schoolhouse is a place where instructors, trainees and training materials are brought together to create a training experience. The weakness of the physical schoolhouse is the requirement that these resources be physically collocated, hence the expenses associated with the collocation process (e.g., travel costs, opportunity costs of soldiers being away from their units). The strengths of the schoolhouse are the availability of a professional instructor, the pool of training resources typically found in the schoolhouse and the opportunities for trainees with a common training need to work together.

The weaknesses in the schoolhouse model are more than economic. There are some limitations in the training benefits produced under the schoolhouse model.

1. Skills may perish before they are needed
2. Training tends to be generic and not adapted to individual unit training needs
3. Trainees may train with others with whom they may never work, hence miss an opportunity to develop a team chemistry or group problem solving approach with the people they will actually have to perform with.

The goal of the present project is to preserve these strengths that the schoolhouse has to offer while making up for its shortcomings—both economic and limitations in the training benefits provided.

We discuss some of the technologies that the Army has used for training, including their strengths and weaknesses.

Simulators and simulation-based training play a prominent role in Army training. There are three types of simulation-based training: live, virtual and constructive. We discovered, through briefings at Ft. Benning and by attending several Army-sponsored training conferences (e.g., the Standard After-action Review System (STAARS)), that the Army's training plan calls for the use of all three types of simulation. The Army has a stated need of having the simulations work together to complement each other in training.

Simulators offer several strengths in delivering training. High fidelity, virtual simulators can create very realistic problem solving scenarios. Networked simulators, including those that use a distributed interactive simulation (DIS) or high level architecture (HLA), offer team training. This addresses one of the weaknesses we cited concerning schoolhouse training, namely trainees may train with soldiers with whom they may not work with when they return to their units. Simulators that support team training can allow soldiers from the same unit train together, thereby allowing them to gain experience with members of the team they may very well be asked to fight with.

Having simulators directly in the field also supports "just in time" training. This allows training to be delivered when needed. If used judiciously, such training can help soldier skills from degrading.

There are some deficiencies in simulator-based training. Some relate to their cost-effectiveness, others relate to their pedagogic value. We address each of these in turn. First, simulators, particularly high fidelity ones, are costly to build. In surveying the simulators used at the Dismounted Battlespace Battle Lab (DBBL) at Ft. Benning, the simulators are built by separate contractors with seemingly independent development efforts. There appears to be little reuse in technology components across training systems (e.g., the Close Combat Tactical Trainer (CCTT), JANUS, the virtual individual combatant (VIC) simulators). Thus, as training requirements change and evolve, the Army seemingly commissions new development efforts.

The present project attempted to demonstrate an approach to software reuse to reduce such development costs. Part of the effort of the present project (which we discuss later), was to develop generic technology components that could be transferred to the present development effort, reused and enhanced. The result was to deliver to the Army training technology that had even greater capability than its predecessor technology, yet cost a fraction of the original development costs to create.

There are also pedagogic drawbacks to the use of simulators in training compared to what a schoolhouse has to offer. In the schoolhouse, trainees receive instruction under the watchful eye of a qualified instructor. Many simulator-based training exercises also have an instructor or observer/controller (O/C) lead the exercise and provide the trainees with feedback.

While the presence of an O/C can go a long way to offset the loss of a schoolhouse instructor by moving training from the schoolhouse to the field, it defeats some of the cost savings realized in the process. O/Cs, like a schoolhouse instructor, must be paid to conduct the training.

There are other problems in using O/C's to guide simulator-based training. First, some of the convenience of "just in time" or "on demand" training is lost as O/C-led training can only be conducted when there are O/C's available. Second, in many team training environments, it is difficult for an O/C to provide trainees with one-on-one instruction. This problem is exacerbated by the fact that advances in networking technology enable greater numbers of soldiers to train together. As the trainee/instructor ratio increases, the availability of one-on-one instruction

declines. Research by Bloom (1984) suggests that there is a direct link between availability of one-on-one instruction and overall performance. Specifically, students receiving one-on-one instructor, on average, perform two standard deviations higher than students taught in a group setting. While the Army can address this problem by increasing the number of instructors or O/C's assigned to each training session, this adds to the costs and defeats some of the cost-cutting objectives of moving training from the schoolhouse to the field. Third, and perhaps most relevant, research by Gentner, Cameron and Crissey (1997) suggests that many O/C's, while competent subject matter experts, are poorly trained instructors. The implication of this is that when the Army does move instruction from the schoolhouse to the field, the cost savings and increase in timeliness of training may be offset by a decline in quality of instruction.

One solution to this challenge is to embed teaching mechanisms directly into the simulator itself. By "teaching mechanisms", we mean the functionality that an instructor would bring to a training session, namely a conjunction of:

1. Knowledge of what needs to be taught
2. Ability to assess the trainee's level of knowledge
3. Ability to tailor the instruction based on what needs to be taught and what the trainee already knows.

The conjunction of these capabilities is typically referred to as an intelligent tutoring system (cf., Brna, Ohlsson and Pain, 1993; Greer, 1995). Earlier we referred to several simulation-based training systems at the Ft. Benning DBBL: CCTT, JANUS, and the VICS. To the best of our knowledge, none of these systems have the conjunction of the above three capabilities. There is empirical evidence that intelligent tutoring system (ITS) technology can fill in when an instructor is absent and produce some of the training benefits shown by Bloom (1984) that occur when trainees are exposed to one-on-one instruction. For example, students having 20 hours of instruction from Sherlock, which teaches avionics troubleshooting (Lesgold, Lajoie, Bunzo and Eggan, 1990), performed on the evaluation test comparably to technicians with almost 4 years experience. Students using the LISP tutor (Anderson, Farrell and Sauers, 1984) to learn the computer language LISP learned the material in half the time it took traditional classroom students.

The present project used ITS technology in an attempt to overcome the lack of good pedagogy found in many simulator-based training systems.

There is another, somewhat related, problem that occurs as training moves to the team environment. Often as exercises scale up to higher echelons, there are more scenario entities than there are humans to play the role of these entities. In such cases, two solutions are possible. The first is to have the computer play the role of these entities. These cases are referred to as computer generated forces (CGF) in the military and intelligent agents (IA), more generically. The other solution is to have a human operator play the role of these multiple entities. A hybrid solution is to use a mixed strategy such as in the case of semi-automated forces (SAF).

As exercises become larger and more comprehensive, the need for human operators will increase in all but the IA scenario. Therefore, advances in IA technology promise to provide the military with additional cost savings in the DIS and, more generally, simulation-based training paradigms. Anecdotally, however, it is not uncommon to hear reports from soldiers, who have received training in simulators that employ intelligent agents, that the intelligent agents are typically not very "intelligent". They report that it is easy for them to distinguish their behaviors from those of humans. As a result, using intelligent agents runs the risk of decreased realism in the training exercise. Since the strength of simulators is their ability to provide realistic training, this sacrifice is significant. By the same token, this need creates a very real opportunity for the present project to develop intelligent agents that exhibit behaviors very close to those of their human counterparts.

We have summarized the strengths and weaknesses of networked training as allowing groups of people to train, but at the cost of reduced personalized instruction and realistic scenario agents. There is another significant strength to networked training that is worth mentioning. Often in networked training exercises, soldiers are not collocated. In other words, soldiers can receive training without the requirement that they be brought together. This is consistent with the stated goal of reducing training costs by bringing the training experience to the soldier rather than requiring the soldier to come to the training.

This basic logic of bringing trainees together via technology rather than physically has formed the basis of a prominent trend in training: distance learning. Here the goal is to bring training to soldiers (or other trainees) by broadcasting the training content. Typically, in distance learning, instructors or training content is kept at a central location and broadcast to remote sites. Such broadcasts can occur via a variety of media: video, teleconferencing, computer networks, etc.

There are some drawbacks with the way many distance learning paradigms are currently implemented and which the current project intended to address. First, the "just in time" goal is often sacrificed as the administrative requirements of assembling the large numbers of students to receive training at remote sites often precludes delivering instruction to individual students when they most need it. Second, distance learning paradigms often sacrifice some of the most beneficial features of the schoolhouse model, namely the opportunity to receive direct mentoring from the instruction and the opportunity to work in collaborative problem solving environments. This is often an unfortunate artifact of distance learning paradigms that simply send teaching by telling over a channel. We do note that there are distance learning approaches that involve instructors interacting with trainees at a distance. In such cases, the goal typically is to have an instructor reach more trainees than s/he would normally do in a classroom. For example, instead of an instructor teaching a single class, s/he will teach 10 classes by broadcasting the instruction to each of these. Often, two way communication is set up so that trainees can ask questions. However, it is clear that the opportunity for trainees to receive personalize instruction when s/he is part of 10 classes of trainees is greatly diminished than when s/he is part of a single class.

One medium over which distance learning is enjoying increased popularity is the Internet. The Internet is designed as a collaborative medium. Therefore, it lends itself to more of the mentoring types of activities typically found in a schoolhouse. The Internet is enjoying increased use in distance learning. One class of technologies that have been used in Internet-based distance learning is Multi-user Dungeons (MUDS) and MUDS Object-oriented (MOOs). MUDS are exploratory environments (typically text-based) that allow users to enter and explore "rooms" in cyberspace. Users can communicate with others in the MUD. A MOO is similar to a MUD but here users are allowed to construct their own rooms and objects (cf., Curtis, 1993). MUDS have become so popular as an Internet-based learning environment that there is even a university (Diversity University) that exists entirely in a MUD (i.e., all courses are taught in the MUD). Many other universities routinely offer Internet-based courses to supplement classroom ones.

We see two critical weaknesses that need to be addressed by Internet-based distance learning technologies such as MUDS and more general distance learning paradigms that center around broadcasting lectures to remote sites.

The first weakness is the limitation of the types of activities that students are exposed to. For example, MUDS foster text-based interaction. Learning activities center mostly on discussion. While current research activities are looking to add multimedia capabilities to MUDS, these efforts are still in the beginning developmental stage. Similarly, while conventional distance learning technologies broadcast more rich media such as video to students, the opportunity for students to interact and engage in learning activities (as opposed to passive viewing) is limited. We believe that distance learning technologies need to provide students with the opportunity to engage in realistic collaborative exercises as they would in a schoolhouse or field exercise. The marriage of Internet-based distance learning with simulation-based instruction would represent a tremendous value added leap in training technology.

The second weakness is the limitation of teacher mentoring. While technologies such as MUDS support teacher participation, one of the unique benefits of MUDS (or the Internet in general) is its continual availability regardless of physical location, time zone, etc. Students routinely use MUDS and the Internet without teacher supervision. To maintain the strength of this availability (which is key to the notion of achieving "just in time" training--i.e., training on demand), it is necessary for the training environment to be effective even when a teacher is not present. Traditional distance learning paradigms suffer from this even more in that they are exclusively dependent on a teacher to broadcast training content.

The technologies that we have discussed so far, along with their strengths and weaknesses are summarized in table 1 below.

Table 1: Training Technologies: Strengths and Weaknesses

Technology	Strengths	Weaknesses
• Simulation	• Realistic • Site-based	• Lacks pedagogic model • Little technology reuse across projects
• Distributed Interactive Simulation	• Team training • Site-independent	• Personalized instruction greatly diminished • High need for non-human agents
• Distance learning	• Site-independent • Supports instructors	• Personalized instruction greatly diminished • Realistic exercises hard to implement
• Intelligent agents	• Reduce manpower requirement for exercises	• Not realistic
• Semi-automated forces	• Greater realism than IA	• Higher manpower requirements than IA

Each of these technologies have historically shared another, important weakness. Typically, each of these technologies is handcrafted from scratch by highly skilled technicians. Once constructed, they tend to be inflexible. As new training requirements are developed, it becomes difficult to modify the technology in order to adapt to these new requirements. As a result, the technology quickly becomes outdated and new technology is created to replace it. This makes the cost of using technology much greater than it could be if there were a way to make this technology more flexible.

We believe there is a common theme as to why these technologies are so inflexible. In many cases, these technologies lack an underlying pedagogic model. By pedagogic model, we mean a model of what the student needs to learn and how to teach it that is independent of the medium through which the training will occur. Human instructors are pedagogic experts. Their subject matter expertise is of the domain rather than of the scenarios that will be used in training. Their expertise in how to teach is based on knowledge of students not system architectures. Therefore, a human instructor can bring his know of the domain into a classroom, assess his students and create instructional exercises from the two.

The same logic applies to how the instructor/subject matter expert would act as a participant in the exercise. The subject matter expert's knowledge is generic. Therefore, she can solve virtually any problem that is consistent with what she knows about. All she needs is information about the problem and how the environment gets updated based on the actions she takes. From this information, she can implement plans and modify them as necessary.

Most training technologies do not operate in this way. Rather, the domain knowledge is coupled with the simulator because this knowledge is used to run the simulated events. Hence, as the training scenarios change, both simulator and domain knowledge need to be rewritten. Because they lack this independent pedagogic model, any training mechanisms they use must be embedded directly into the simulator and the scenarios used. In other words, because the technology will not assess the trainees and teach them based on pedagogic expertise, the simulator must have preprogrammed rules about how what scenarios will be presented and how material will be presented so as to instruct the trainees using the technology.

As training requirements change, these built-in training approaches will no longer be applicable. However, since they are an integral part of the technology's design, it is typically more cost-effective to rebuild the simulator than to modify it. Unfortunately, because the core technology behind the training approach is no longer applicable, there is little reuse of this technology. This coupled with the fact that supporting technologies such as graphics capabilities are rapidly improving, it makes little sense to try to reuse these portions of the simulator as well.

The same logic applies to computer generated forces (CGF). If the CGF are driven by rules that tell the agents how to act in a specific scenario, then once that scenario is outdated, the rules are no longer appropriate. Hence, the CGF needs to be rewritten as well.

This discussion on how the design of these technologies contributes to early obsolescence makes an important point. In designing training technologies, it is not only important to select the appropriate technologies today so that the resulting system will exploit the strengths of the technologies while overcoming their weaknesses, but it is also important to design and implement these technologies with a view to future training requirements. This design must become more generic and modular so that as training requirements or technology capabilities evolve, the training system itself can be enhanced without having to be discarded and rebuilt from scratch.

The Present Project Approach

Before discussing the specific objectives of the present project, we summarize the original military training goals, relevant technologies and strengths and weaknesses. The relevant goals are:

1. Reduce costs
2. Provide "just in time training"
3. Provide realistic, high quality training
4. Provide local control for adaptation of training to unit needs
5. Provide modifiable training for evolving needs

Technology is seen as an opportunity for supporting all of these goals. Specifically, simulators are useful in reducing costs by bringing training to the soldier. Because they come under local control, training can be delivered "just in time". A high quality simulator will present realistic, high fidelity scenarios.

DIS builds upon standard simulators by networking them and allowing multiple trainees to train together on team tasks, thereby increasing the realism and value of the training experience. These tend to create an additional requirement for additional scenario entities, which can result in increased costs if humans play these roles or decreased realism if a computer plays them. Also, DIS relies on the same simulators discussed above and therefore, suffer the same inflexibility. Because multiple trainees are participating, the lack of human instructor time is exacerbated.

Distance learning is a superset of DIS. Many instantiations support instructor interaction. However, when this occurs, it is usually in the context of multiple classrooms thus reducing the amount of attention per soldier. Also, in such cases, the emphasis is on lecture-based instruction rather than practical problem solving exercises such as found in the simulation world. This also reduces the realism of the instructional experience.

CGF and SAF are ways of trying to fill in when more scenario participants are required than humans are available. CGF is the most economical but suffers from lack of realism. SAF is more realistic but is less economical. CGF has the least flexibility as there is no human intervention. SAF is flexible within the parameters offered to the human operator to manipulate.

Collectively, a combination of networked simulation that can be delivered at a distance, coupled with quality CGF offer the best combination of meeting Army training objectives. Networked simulation offers the opportunity for realistic team training. Distance learning offers the ability to bring this training to local sites for "just in time" training in the most cost effective manner. CGF support team training by allowing as many trainees as are available to train regardless of how many are needed to perform the scenario realistically.

However, in order to utilize these technologies in a way that do not succumb to their respective weaknesses, the following four enhancements need to be made.

1. Pedagogic models need to be embedded in the simulators to provide trainees with one on one instruction, regardless of whether human instructors are available
2. Intelligent agents are needed that act as realistic scenario participants so that trainees can train "on-demand" without regard to how many trainees are actually available
3. Scenario editors are needed to allow customization of training to unit needs
4. The technology needs to be designed as generic and modular as possible to allow for modification as training requirements change.

The remainder of this paper discusses the results of a Phase I Small Business Technology Transfer (STTR) project in which we integrated and enhanced several component technologies in an effort to address the above objectives. Because the goal of the STTR program is technology transfer rather than the development of a brand new technology, we chose the Dismounted

Infantry (DI) military operations in urban terrain (MOUT) as the environment for our Phase I testbed. MOUT was the training domain we had established in previous projects as an area of high importance to the U.S. Army Infantry. In particular, we focused on developing training technology for squad leaders and fire team leaders performing the task of clearing a building.

In performing the project, our first step was to collaborate with members of the Ft. Benning Infantry community. Our objective in this collaboration was to identify training needs that would form the basis of project objectives and then have the members of the infantry community review in-progress developments of the technology. We are grateful to the following members of the Ft. Benning community for their help and support on the project: the Directorate of Operations and Training (DOT), U.S. Army Infantry School (USAIS); the Battle Lab and the U.S. Army Research Institute – Infantry Forces Research Unit. Through these interactions, we established the following project goals for our technology.

1. Training problems should be realistic
2. Instructional mechanisms should be embedded in the training environment such that the technology could be used with or without an instructor present
3. The technology should support individual or team training regardless of whether sufficient participants are available
4. The training environment should support customization of training scenarios to unit objectives and therefore be under unit control
5. The technology should be generic and modular to support modification and reuse as training requirements change.

The Virtual Classroom Technology

We believe the key value added contribution that can be made to the simulation-based training world is to make the learning environments "intelligent". By intelligent, we mean that the environments should embody the same sorts of skills a live teacher would have, were there one always available to work on an individual basis with the students. What teachers bring to the process is a knowledge of what needs to be taught, an ability to assess students to determine their learning needs and styles, a knowledge of how to teach (including what exercises are necessary) and an ability to get the training resources necessary to deliver those exercises. In order to make the learning environment intelligent, we integrated intelligent tutoring system (cf., Brna, Ohlsson and Pain, 1993; Greer, 1995) and intelligent agent technology into the learning environment. This is manifest both in terms of implicit intelligence in the activities that the trainees perform and in the form of intelligent agents that can serve as mentors and co-problem solvers when needed to reduce the requirement that instructors and other trainees always be available in the virtual schoolhouse. Ultimately, intelligent agents may serve other roles such as "reference librarians" to help locate material or as sources of subject matter expertise that a trainee could query for information. These latter capabilities are beyond the scope of the current effort but are part of RDC's overall strategic plan for the Virtual Schoolhouse.

Based on this, our goal was to develop a simulation-based intelligent tutoring system (SITS) to provided realistic problems with embedded instruction and integrate intelligent agents into the simulation to provide realistic human behavior for computer-generated forces. We also developed a scenario editor which served two functions: it allowed scenario customization and it demonstrated the power of our technology by showing that it would work effectively on scenarios generated by end users, not just those generated by the project team.

The Phase I technology was based on transfer and enhancement of technologies developed under two related projects: a project sponsored by STRICOM to develop a squad and fire team leader MOUT trainer using a virtual simulation-based intelligent tutoring system and a project sponsored by the Defense Advanced Research Projects Agency (DARPA) to develop intelligent agents for MOUT.

The core technology that was common to these projects was an expert system model based on an integrated knowledge structure (INKS) framework. The concept behind INKS is to model the knowledge an expert would have about the domain. This knowledge model could serve several purposes. To implement a pedagogic model, the INKS would assess whether the trainee has the desired knowledge to master a task and what his learning needs were. To play the role of an intelligent agent, the INKS would compute a solution that an expert would use which could then be carried out in the solution. To support a generic architecture, the INKS could be made as a generic problem solving model, which is how experts understand their domain. In this way, the INKS is independent of the training delivery system and can be modified and reused without disrupting the basic integrity of the system.

Because INKS is central to our technology, we describe it in detail below.

An expert knowledge model of MOUT

In the cognitive science and psychology literatures, several frameworks have been proposed as models of expert (and non-expert) knowledge. These schemes tend to address different types of knowledge. For example, scripts (Schank, 1982; Schank and Abelson, 1977) are used to represent goal and planning knowledge that is used in fairly routinized environments. Scripts are generalized sequences of steps used to achieve a goal. Script-like schemas can also be used to integrate bodies of knowledge into a larger framework.

Knowledge about data patterns and how objects are organized together (e.g., the configuration of a laboratory) can be represented by object frames (c.f., Anderson, 1980; Minsky, 1975). Frames are very much like scripts in that they are expectancy-driven organizers of knowledge. We conceptualize scripts as focusing more on goal and plan-related knowledge while frames organize collections of objects. Frames can also be distinguished from semantic nets (cf., Quillian, 1966) which tend to organize information about individual concepts and relationships between them rather than collections of objects. For example, a fire team may best be represented by a frame since it is a collection of people and equipment while a rifle may best be represented by a semantic net that describes its features.

Knowledge about situation-specific procedures can be represented by production rules (cf. Newell and Simon, 1972). Production rules are expressed in the form "IF [antecedent], THEN [consequent]", where antecedents are situational conditions that determine when procedures are to be executed and consequents are the procedures executed under those conditions. Production rules are useful in both carrying out procedures (e.g., "If this step has been completed, then do this next step.") and also generating inferences (e.g., "If the following problem features are observed, then infer that this is an [X] type of problem."). Production rules can be distinguished from scripts in that scripts organize entire goal-driven plans, while production rules organize specific actions. Scripts can be viewed as collections of production rules much the way that frames can be viewed as collections of semantic nets.

Finally, causal and analogical reasoning can be captured by mental models (cf., de Kleer and Brown, 1981; Johnson-Laird, 1983; Leddo, Cardie and Abelson, 1987). In our framework, (Leddo, Cardie and Abelson, 1987), mental models are viewed as encoding the causal rationale for why a specific problem solving procedure is used. One of the factors that distinguishes the way experts solve problems from the way non-experts do is the former's heavy reliance on mental models and the ability to use them to select an appropriate problem solving strategy to meet a set of objectives.

We have discussed five different representation frameworks (scripts, object frames, semantic nets, production rules and mental models) for representing expert knowledge. Experts possess diverse knowledge that is richer that can be handled by any single framework (Leddo et al., 1990). Leddo, Cardie and Abelson (1987) developed an Integrated Knowledge Structure (INKS) framework that combines these individual schemes. In the INKS framework, scripts serve as the general organizer of knowledge, linking plans and goals together. Production rules give situation-specific procedures to be executed given conditions that arise during the execution of a plan. Frames organize collections of objects that are utilized in the execution of plans while semantic nets organize features of the individual objects within a frame. Mental models provide the rationale for why procedures are executed and how they are instrumental in achieving objectives.

The INKS framework can be used to model MOUT knowledge. For example, a script could represent the process of clearing a building such as breaching the building, moving through hallways, stacking, breaching doorways, clearing rooms, etc. Production rules could model specific steps in the overall script such as how a four man fire team should enter a room with a center door. Semantic knowledge could model knowledge about specific equipment or building features (e.g., closed, unmarked door with 2 hinges and 1 lock). Frame knowledge could model knowledge about the building and its layout. Mental models could model the rationale behind certain actions such as why it is important to provide security before sending a fire team down a hallway.

The INKS framework served as the basis for modeling the domain, the trainees' actions in the simulation environment and determining how to update the simulation. The expert model also served as the basis for driving the behaviors of the intelligent agents.

Technologies Transferred To The Present Project

STRICOM 1-Person Tutor.

RDC built a one-person simulator with embedded intelligent tutoring for STRICOM. This simulator consisted of three primary modular components: a 3-Dimensional Simulation Module (SIM); a Case-Based Scenario Definition and Control Module (CBR/CTRL); and an Expert System (ES) with Error Detection Module. The training domain of the system is doctrinal tactics for Fire Team Leaders clearing buildings in close-combat on urban terrain. The architecture of this system is presented in Figure 1 below.

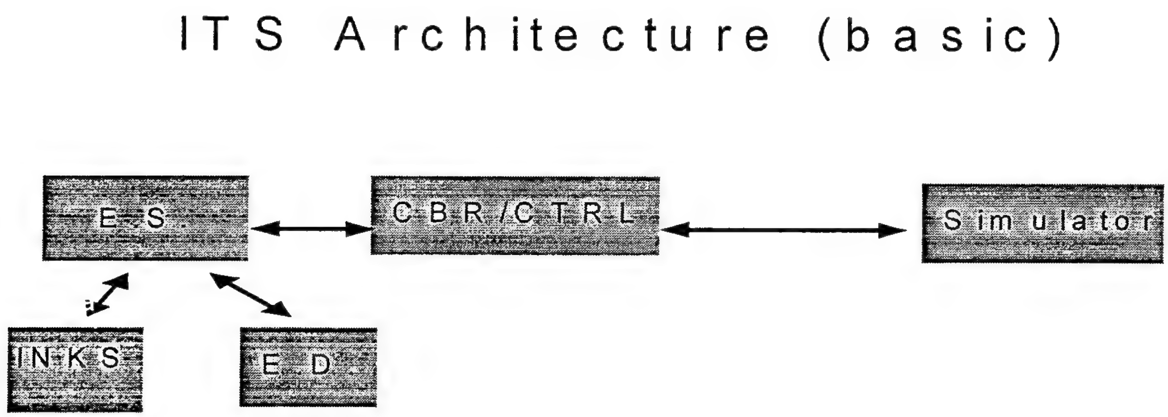

Figure 1: Basic ITS Architecture

Each of the three modules is self-contained. The SIM module is responsible for handling most of the interface between the trainee and the system. Driven by STRICOM's requirements, the original SIM presented a 3-dimensional virtual environment on a flat screen. The trainee viewed the world on this screen from the perspective of the Fire Team Leader. To focus our efforts on the building of a simulator that could assess and plan remedial activities for the student, we did not burden our efforts with reasoning about free-form roaming in the virtual world. Rather, the student Fire Team Leader could indicate his/her intentions by issuing a keyboard or verbal command that caused the Fire Team to take specific actions in the virtual world.

The CBR/CTRL module is responsible for two separate functions. The primary function is to serve as the director of the training exercise. It monitors student's progress in the course and adapts the remedial activities to those which seem to provide the most benefit to the student. Ultimately, the CBR will select the appropriate training scenario from amongst many to present next for a given student, based the system's assessment of the students needs. The secondary function performed by the CBR/CTRL module is that of "middleware". – It serves as a bridge between the simulated virtual world and the ES. The middleware achieves this by –

- Reporting changes in world states in the SIM (including commands issued by the Student) to the ES and,
- Effecting world changes requested by the ES in the virtual world of the SIM by queuing world state function calls to the SIM.

The ES is engineered using RDC's INKS theories to drive its development. The INKS module contains generic knowledge about the tactics of a Fire Team in clearing rooms in urban combat. The source of this knowledge was a combination of US Army Doctrinal Manuals and Subject Matter Experts provided by the U.S. Army Infantry School, Fort Benning, Georgia. The knowledge put into the ES is generic. It is not specific to any particular battlefield terrain. The ES performs two key roles in the STRICOM system. First, it reasons about the "correctness" of an entity's action in the SIM. We use the word "entity" because there are actions taken by the student as the Fire Team Leader and there are actions taken by the remaining Fire Team members and OPFOR in the simulator. The ES monitors the appropriateness of each of these. The second role of the ES deals with Error Diagnosis. The ES attempts to determine possible reasons to attribute to why an entity has made an error. The ES forwards these errors' attributions to the CBR/CTRL, who schedules appropriate remedial training.

In summary, for the STRICOM 1-Person Tutor, we developed a modular solution such that the SIM (simulator), CBR/CTRL (case-based reasoner/controller) and ES (INKS-based Expert System) combined to provide the student with an immersive learning environment. This environment tailors itself to the needs of the student and provides an instructor-in-the-box for the student. We have a rich training environment with an instructor – one of the overall goals of this project.

DARPA Intelligent Agents.

RDC built a 2-Dimensional Virtual or Constructive Simulation to demonstrate how an INKS-based Expert System (ES) provides flexible training opportunities to the soldier. A principal concern during this project was to develop intelligent agents (IA) whose behavior in a simulation closely approximates that of human-controlled agents.

By design, RDC built this system using components already designed and developed for the STRICOM project described above. There were, of course, some modifications. We briefly describe each of these, using the terminology introduced above.

RDC replaced the entire SIM. Whereas the STRICOM effort called for a 3-dimensional, immersive presentation, this effort only needed a birds-eye 2-dimensional representation. However, since there was a well-defined interface between the CBR/CTRL and the SIM, the design and coding of the SIM were very straight-forward.

The CBR/CTRL needed nominal modifications. Remedial assessment was not part of the IA effort. For the most part, the effort required disabling much of the existing functionality of the CBR/CTRL Module. To allow for the fact that we introduce scenario editing capabilities to the SIM, we modified the control aspects of the CBR/CTRL.

The ES needed no specific modifications. The generic structure and contents developed for the STRICOM were usable as designed in STRICOM. What did occur was the addition of some new knowledge concerning both enemy soldier behaviors and to demonstrate friendly reaction to changes in the situation in the simulator. A side effect of this effort was that we exercised much more of the knowledge base in the DARPA project. This allowed for our identifying and correcting knowledge representation errors made in the STRICOM effort.

RDC used two approaches to demonstrate “agency”. One entails reflex responses to environmental factors. The other entails a more complex cognitive model, using INKS. The reflex responses are those which occur when some global state in the SIM world change. For instance, when the OPFOR hear gun fire, they begin to position themselves into better defensive positions in their respective spaces. Another example of this reflex response is that of the Fire Team using Flash-Bangs. The friendly force works in silence until it comes across an enemy. Then it uses the noisy Flash-Bangs to add additional margins of safety to their clearing rooms.

We briefly describe how these “new” capabilities affected the ES. In the case of the OPFOR moving around once they are aware of the friendly force presence, we simply added new knowledge to our ES. This is the beginning of building a separate ES for OPFOR in a training exercise. In the case of the Flash-Bang knowledge, we needed to modify our existing knowledge base about doctrine concerning room clearing. We added a production rule that said, “If we have come across and enemy, then do a Flash-Bang Procedure." Adding this knowledge to our ES was trivial. Our Fire Teams now know when it is appropriate to use a Flash-Bang.

Of course, complex behavior of agents was the goal of this project. The strength of our INKS approach to the STRICOM ES pays off on the DARPA project. Recall that in the STRICOM project we needed to determine if the action taken by an entity was appropriate given the situation. If not, some remedial action was required. By using the very same mechanism, the ES determines what is the most appropriate action given the current situation. By reversing the flow of information, the ES can tell the CBR/CTRL that entity X should do Y next. The CBR/CTRL relay this to the simulator. This mechanism provides an autonomous IA that reacts to the situation it finds itself.

To demonstrate our capabilities, we also needed to build a scenario editor. It was clear from our discussions with members of the Infantry community that multiple training scenarios were needed. Both to meet this objective and to test the power of our expert knowledge model, we constructed scenario editing tools that allowed real time changing of simulation parameters.

This feature was supported entirely by our modular architecture. If the expert knowledge model was scenario dependent, then any change in the scenario would throw the model off. However, because our architecture was constructed such that the expert knowledge model learned of the scenario as the trainee progress through it, it did not distinguish between what was a preset scenario feature or one that was created on the fly.

The scenario editing tools were created based on parameters the underlying expert knowledge model would understand. This encompassed a wide range of parameters. Among these were:

- Number, location, training level, combativeness of the enemy
- Number and location of civilian non-combatants
- Speed and accuracy of friendlies and enemies to fire their weapons
- Weapon lethality
- Whether doors were open/closed, marked/unmarked, and how many locks and hinges they had

As we discussed earlier, our system uses a modular architecture. As a result, the expert model that runs the intelligent tutoring system and the intelligent agents receives information from the simulation in real time. It has no prior knowledge about what the scenario features are. Therefore, scenario features can be changed in real time without disrupting the flow of the simulation.

In this case, we could alter the situation and evaluate whether our agents behaved appropriately to the situation. In addition, we needed ways to have the system act autonomously and compare it with agents controlled by humans. We developed a mechanism whereby control of an agent could toggle between human and the ES.

Summarizing the DARPA IA project, we built a 2-Dimensional Constructive and Virtual Simulator, depending on who was controlling the agents. The system used a significant section of the middleware developed for the STRICOM project and made extensive use of the existing INKS-based Expert System. We added capabilities to edit the scenarios and to vary the situations faced by the ES. Finally, we added the capability to allow the control of agents to be either human or machine. These capabilities along with further enhancements to the STRICOM project in its second year are much of the technology transferred in this project as described below.

PHASE I VIRTUAL SCHOOLHOUSE

The idea of a Virtual Schoolhouse immediately conjures up need for a network. The Army has invested millions of dollars in the DIS and HLA simulation technology to allow for "networked" training environments. In the current project, we address several issues that arise in networked training systems. Some of solutions exist in DIS and HLA compatibility.

Training an individual Fire Team Leader in proper Fire Team tactics in MOUT operations is of value to the Army. However, the Fire Team Leader is but one of many levels of command and control involved in such an operation. The need to have simulations with embedded instructors that could scale up the echelons where higher levels of cognitive decision-making is self-evident.

To demonstrate our approach, we developed a two-person, training simulation that transferred much of the technology from both the STRICOM project and the DARPA project. Because the current effort was under a Phase I STTR contract with limited funds, we build this system using the 2-dimensional simulator developed during the DARPA project. The system could have easily have been built upon the STRICOM (1-person trainer) project. This would have required additional funds.

As it turned out, to complete our STRICOM project in the second year, many of the technological problems solved in this effort also applied to the completion of that effort. In other words, we had bi-directional technology transfer on this project.

In the following paragraphs, we describe the Phase I system built during this project. Again, we use the already introduced terminology concerning the various modules of the system.

A major difference between this effort and the prior efforts is that we now are using two machines to train two people simultaneously. We reviewed several topological solutions. We rejected a client-server architecture because we believed it would make it more difficult to become DIS compatible in the future. We selected a peer-to-peer network using TCP/IP. This turned out to be a good selection and the basic networking capability was rather straight-forward to develop.

Figure 2 below shows the architecture of the networked two person ITS:

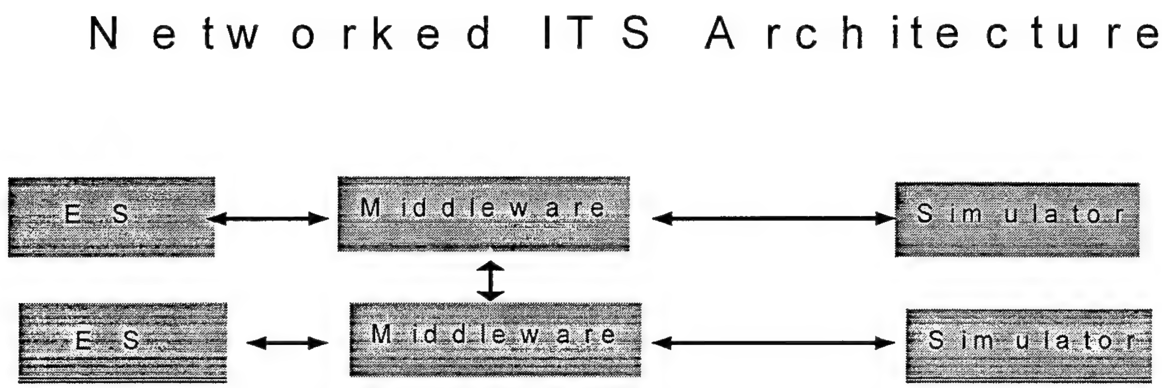

Figure 2: 2 Person Networked ITS Architecture

While this provided the basic capability needed to build our demonstration system, there remain many of the problems native to distributed simulations. Some were not bothersome in the Phase I demonstration. We defer others to the Phase II system development. For instance, we assume a network latency time of zero, but the effect of this is negligible because we are using real-time clocks on each simulator that are not that much out of synchronization. Likewise, we are using redundant modeling to simulate the behavior of the "computer-generated" entities. We realize that as we scale up we will need to have more centralized control of these entities as is envisioned within the HLA.

The SIM module for this demonstration is the same as was developed for the DARPA project. There are simply two copies running.

The CBR/CTRL module required the most significant changes for this effort. First, it needed a capability to communicate with another CBR/CTRL module. We developed a simple means to broadcast messages back and forth between the two simulations. The CBR/CTRL module continues to handle the traffic between the SIM and the ES. It also sends "user did this" information to the other computer. The CBR/CTRL also receives messages from the other computer and routes it to the SIM or ES as appropriate.

The ES system is the same as DARPA with one exception: commands that were issued by the other machine are filtered out and ignored except to update appropriate world state information. Thus, the INKS representation developed originally for the STRICOM project, adapted during the DARPA project and used in the Virtual Schoolhouse application are the same with minor modification. Thus, one knowledge engineering effort led to a variety of products.

We also adapted the scenario editor and agent technologies from the DARPA project. They both required significant re-work to handle synchronization issues between the two machines. The final solution allows for run-time modifications of the scenario such that the modifications occur simultaneously on two platforms.

We highlight agent technology. One of the problems with training systems that require several simulations running concurrently, is what do you do when you have fewer participants than you have simulations? You can either wait until you have enough human participants or you make use IA technology to have computer-generated entities simulate the other participants. We have demonstrated the ability to change the number of human participants in a distributed simulation dynamically.

In summary, we took technologies developed under separate efforts – STRICOM and DARPA and have integrated them into a more flexible training system for the Army. We believe that the key to success lies in keeping the ES, the SIM and the middleware separate.

Evaluation Results

We evaluated our technology in two ways. First, we evaluated the realism of our intelligent agent technology with military experts. Second, we had military experts review our technology to provide us with feedback for future directions.

Evaluation of realism of intelligent agent technology.

There were two forms of this evaluation. First, in several demonstrations of the technology, one at Research Development Corporation and several at Ft. Benning and at STRICOM, we had military experts create their own scenarios, both at the start of a simulation run or in real-time during the execution of a simulation-run. In all cases, RDC personnel had no prior knowledge of what parameters the experts would wish to change. In no case, did the system fail to operate (often referred as "crashing"--a common demonstration phenomenon when a system is given input outside its working parameters) or exhibit an unexpected behavior, suggesting it was incapable of coping with the scenario.

The second form of the realism evaluation was to pre-record both a human and the system controlling the fire team leader in separate runs of identical scenarios. Both pre-recordings were played side by side for groups of observers. There were three groups of military experts (numbering approximately 6-8 per group), who would be familiar with the expected behavior of humans working in the scenario and three groups of engineers (numbering approximately 2-10 per group), who would be familiar with behaviors generated by expert systems. Again, one group was conducted at RDC, two at Ft. Benning and three at STRICOM. The experts were asked to distinguish which recording was made of the human operator and which of the system. In general, the experts and engineers were unable to make the distinction (many even guessed wrong). Those that did guess correctly cited a shorter latency between computer-issued commands than human-issued ones, presumably because the human needed to enter commands through a keyboard, which took time.

This evaluation raised an interesting additional benefit for our technology. Experts occasionally found themselves disagreeing with each other as to what the "correct" action should be in a situation. While this did raise the important point of validating the underlying expert knowledge used by the system, it also showed how experts often do disagree with each other, and having an expert model articulate tactical decisions is a means for prompting experts to discuss and reason about their problem solving methods.

Evaluation For Feedback For Future Directions.

In the present project, we worked very closely with members of the Infantry community at Ft. Benning. Upon completion of the project, we asked two distinct communities within Ft. Benning to evaluate our work and provide us feedback. These were the Directorate of Operations Training and the Battle Lab. Below, we present common points that were general across the communities as well as the areas in which their needs differed.

In general, our work was positively evaluated by all who saw it. All agreed that the work was valuable to the Army and should be continued. In presenting their views in how the work should be continued, the following consistent points were made:

1. The technology may be centrally located, but should be locally used and controlled.

Most distance learning technology holds the technology in a central location and then "broadcasts" (e.g., via satellite, Internet) the content to remote locations. This tends to serve two purposes:

- The technology can be updated at one location and disseminated from there, thereby reducing administrative costs of mass distribution.
- The distributed nature of the technology allows soldiers from different locations to train together.

The members of the Infantry community indicated that for the types of training that we were focusing on, that units train together as an integral unit using tactics, techniques and procedures that get adapted to the specific unit. Therefore, while the first benefit of distance learning was applicable, the second actually operated at cross purposes to the way units currently train.

In other words, the Infantry evaluators were of the opinion that there would be very little training with soldiers in different locations. Further, having a single training system that was designed for every unit was at cross purposes with the goal of allowing each unit to customize training to their tactics.

The solution that was proposed was to have a central location for the technology that could be accessed by the different units. However, the desired goal is to allow each unit to download the technology and run it locally, while furnishing them the tools to customize the technology to their specific missions and tactics.

This represents a significant departure to standard distance learning paradigms that may have users access a web site and receive individualized on-line training. This is flawed for several reasons. First, it omits the team training that is central to Army operations. For tasks such as MOUT, the bulk of what needs to be trained is team operations. The standard web-based instruction does not handle this. Second, many distance learning environments are not suitable for team training and will fail when scaled from individual to team training. The Internet is a prime example of this. Team training requires that real-time scenario and event information be made available to all participants in a timely manner. Distributed Interactive Simulation (DIS) networks have as a requirement that transmission can be made in such a manner. The Internet is far too slow and variable in its transmission rates to support real-time team training. The trouble that this runs into is that different trainees can have tactical scenarios that are out of synchronization with each other, thereby influencing what events they think are happening (e.g., one person may think there is an enemy soldier in the same hallway, whereas another might think that the enemy has already left).

The solution we came up with, store the main technology on a central, web-based location, but allow it to be downloaded and run locally, meets these needs. First, the central location allows easy access to the latest updates to the technology, which is consistent with the distance learning intent. Second, having the technology be downloaded and run locally, still moves training from the schoolhouse to the field (thereby realizing the cost savings that distance learning is desired to achieve), but meets the goals of giving the units the opportunity to run the technology on a local network (thus meeting the real-time transmission requirements of team training) and giving units local control of the software so that they can adapt it to their needs. Of course, the latter goal requires that supporting tools be made available so that the software can be so modified.

2. The training be scaled up to focus on platoon leader as well as squad leader and fire team leader training.

The present system allows a squad leader and fire team leader to be trained. Members of the Infantry community expressed a desire to see the MOUT training be scaled to the platoon level. This was so for two reasons. First, it is typically the mission of a platoon to clear an entire building. Therefore, it is the right level of echelon for such training. (It was also noted that the platoon leader was probably less experienced than the squad or fire team leaders and may be in greater need of the training.) Second, the platoon leader's decision making is less procedural than that of the fire team leader or the squad leader. It was felt that for the types of procedures that fire teams and squads carry out, that these procedures can be trained just as well and cheaply using live simulations (e.g., moving in hallways, entering rooms). Platoon leader decision making is far more complex and typically involves decisions that do not lend themselves well to live simulations such as dealing with breaches in buildings, calling in air support and dealing with buildings of different construction. It was generally felt that our virtual schoolhouse technology would be most useful in training these sorts of decisions as opposed to the lower level procedures that characterize fire team leader decision making.

3. There need to be editing tools that allow the users to vary scenario parameters.

The comments from 1 and 2 made it clear that an important direction that we need to take in the future is giving the user community the ability to create their own scenarios. Currently, our technology allows some customization in terms of which doors are open or closed, marked or unmarked, number and location of enemy and civilian forces, whether the enemy are armed or unarmed, well or poorly trained, etc. The two main types of editing tools requested by the Infantry were:

- Tools to vary the floor plan and building construction so that different tactics and theaters may be represented (e.g., the flimsy construction of Somalian buildings compared to the more sturdy construction of European buildings) and
- Tools to allow the user to construct different enemy doctrines and tactics so that they could depict different threats that soldiers might face. Consistent with this was the desire to allow the enemy forces being played by humans as well as agents.

In the present project, we focused primarily on developing constructive simulations. However, as discussed earlier in this paper, much of the technology being transferred in this effort comes from a separate initiative in which virtual simulations are being developed for MOUT training. We found a general preference in the DOT community for virtual simulations and in the Battle Lab community for constructive simulations. This appeared to be directly linked to their respective interests.

DOT was concerned with leadership training. Therefore, members of this community liked the virtual simulation because it was more immersive and more closely represented what a soldier would see in an actual simulation. Additionally, it was observed that the added graphics might make the training experience more fun and "gamelike" thereby inducing soldiers to seek training on their own time.

The Battle Lab was more concerned with modeling and analysis. In particular, they had a strong interest in mission analysis and mission rehearsal. Therefore, they were more interested in seeing how the events of the battle unfolded and liked the constructive simulation where they could see everything. The virtual simulation provided a "trainee's eye view", although there was a toggle that provided a "sky view" as well.

Also the Battle Lab was particularly interested in conducting the "what if" analyses for the different scenarios. Therefore, they were especially interested in having scenario editing tools that allowed them to change different assumptions of the battle and the simulation model.

Conclusions and Future Directions

We cited earlier in the paper the following technical objectives:

1. Pedagogic models need to be embedded in the simulators to provide trainees with one on one instruction, regardless of whether human instructors are available
2. Intelligent agents are needed that act as realistic scenario participants so that trainees can train "on-demand" without regard to how many trainees are actually available
3. Scenario editors are needed to allow customization of training to unit needs
4. The technology needs to be designed as generic and modular as possible to allow for modification as training requirements change.

We believe we were successful in meeting each of these objectives. First, our intelligent tutoring system technology provided the pedagogic models that gave trainees one-on-one instruction. The built-in remediation toggle allowed a user to turn this instructional feedback on or off such that the system could be used with or without an instructor.

Second, the technology had intelligent agents that could play the role of fire team leader or squad leader. Based on our earlier discussions, we noted that the control of these agents could pass between human and computer in real time so that maximal flexibility was provided in accommodating the availability of human participants. We demonstrated two tests of the power of these agents: the scenario editor that allowed real time changes in the scenario and the recorded sessions of human and computer controlled forces that were shown to subject matter experts. We observed no case in which a scenario created by a user caused unexpected behavior by the IA and subject matter experts were generally indifferent when attempting to distinguish IA from human behavior.

Third, a scenario editor was created to support customization of training to user needs. This is perhaps the weakest part of our project, not because the scenario editor is faulty but because it is very incomplete in providing users the opportunity to create scenarios that vary all the parameters they would want varied. Enhancing the scenario editor would be a high priority activity in Phase II.

Fourth, the architecture we created was clearly generic and modular. The generic nature of our expert system technology was demonstrated in two ways. First, the scenario editor demonstrated that the expert system was scenario independent. Second, the fact that it could be transferred from project to project with no change demonstrated both its independence from the final application and its modularity.

The modularity of the technology was also demonstrated in the technology transfer effort. We cited that in the STRICOM project a 3-D simulation was used, whereas in the present system a 2-D version was used. In both cases the middleware and expert system were preserved. This shows that the simulator could be changed in the basic architecture without changing the remaining components. This shows that the simulator was modular.

Also, when we scaled the system from 1-person to 2-person, we created a new expert system for the squad leader but used the same simulator. In other words, we also showed that the expert system was also modular.

More importantly, the present project represents technology that reused and enhanced technology from 2 other projects: a $600,000 effort and a $100,000 effort. The present system's capability was greater than either of the other two, even though it was also developed at a budget of $100,000. We believe this represents the true spirit of creating cost effective technologies that can continually provide the Army with state-of-the-art training technology at low cost. The present system represents an enhanced version of its predecessors at a lower cost. We hope this is a good demonstration of how to improve the training technology development process so as to create maximum value to the Army.

The evaluation results of the present project also suggest that combinations of technologies used in the project (simulation, intelligent tutoring system, intelligent agents) offer promise in moving training from the schoolhouse to the field. We also believe there is extensibility of the technology to other simulation-based training applications, particularly as simulation scales to larger, more comprehensive exercises.

As exercises encompass more and more trainees, the role of the after action review (AAR) will also change. In large scale exercises, the AAR will necessarily emphasize the general performance of the units involved. There will not be sufficient manpower to assess and remediate trainees at the individual soldier level. This is where our intelligent tutoring system technology comes in. The ITS technology offers the Army the opportunity to provide assessment and remediation at the individual solider, small team or unit level, while still preserving the integrity of the large scale training exercise.

A related benefit comes in enhancing the AAR process itself. The ITS performs its assessment by tracking soldier actions against its own underlying expert problem solving model. As such, the ITS is equipped to provide an automated AAR. The Army can use this technology in two ways. First, it can use the ITS to provide AARs that are customized to specific subunits, teams, individuals, etc. that might be missed as part of the overall AAR. Second, the underlying expert model can be used as a basis for singling out key events in the exercise or identifying critical mistakes and areas that went wrong. These events could be passed to the observer/controller who could discuss them in the AAR.

This latter benefit can be responsive to a stated need by the Army to develop a standardized AAR process. Currently, AARs are strongly shaped by the expertise and interests of the O/Cs. As might be expected in any area of human performance, there is likely to be tremendous variability across O/Cs in their ability to assess and remediate trainees. The ITS technology can serve the role of an equalizer to assist O/Cs in this regard.

In addition to the component technologies having benefit outside the scope of the "virtual classroom" concept, the technologies transferred and enhanced in the present project make a unique contribution to a virtual learning environment.

As we discussed in the beginning of the paper, most distance learning environments focus on broadcast of content to a wide audience. In such cases, interactivity, collaborative problem solving and personalized instruction found in schoolhouses are often lost. The goal of the present project was to combine the best of both worlds—provide the widescale dissemination found in typical distance learning environments with the instructional benefits of the schoolhouse.

We believe the combination of networking and artificial intelligence-based technologies demonstrated in the present project offer an opportunity to obtain the best of these two worlds. Networks support both dissemination and collaborative problem solving, while AI supports the interactivity and personalized instruction. A secondary benefit of the AI technology is to provide maximum flexibility in supporting "on-demand" training. This was evidenced in two ways. First, the dynamic transfer of control of agent behavior from human to computer and back supports collaborative problem solving, regardless of whether other trainees are available for collaborative problem solving exercises. This allows trainees to have the flexibility to receive training when it is convenient to them, regardless of the convenience to others. In this way, the technology can be used more often without requiring the administrative efforts of assembling teams of trainees.

Second, the ITS toggle switch supports training with or without an instructor. Therefore, when an instructor or O/C is present, the ITS can be turned off, allowing the instructor to lead the training. However, by turning the ITS on, the trainee is not dependent upon the availability of a human instructor in order to receive full training benefits. Again, training can proceed at the convenience of the trainee, without the requirement that a human instructor be available.

We see this type of training flexibility as a model for future distance learning training environments. Costs can be reduced by eliminating the administrative requirements of coordinating training sessions across all trainees and trainee sites. The AI features eliminate the requirement that other trainees and instructors be present in order for training to proceed.

Training benefits are similarly increased. Specifically, the AI features provide two enhancements. The intelligent agent technology makes the training scenarios more realistic such that the training should be more transferable to real world problems. Also, the intelligent tutoring system technology provides each trainee the opportunity for individual instruction—a feature that is often lost when content is broadcast to remote sites.

This latter point raises a key issue. We have found, in the present project, that it is not only important to customize training to the individual needs of each student, but also to the needs of each unit. Feedback we were given at the Ft. Benning community suggests that units tend to train together on missions, tactics, techniques and procedures that are unique to each unit. Therefore, standard distance learning environments that broadcast the same content to these different sites fail to take this need into consideration.

For this reason, we developed some preliminary scenario editing tools in the present project. Clearly, these tools are far from complete. The primary limitation with the tools is that the scenario parameters and their values were predefined. We identified four primary needs with regard to scenario editing capabilities. These were:

- Friendly mission (e.g., offense, defense, rules of engagement)
- Building characteristics (e.g., construction, layout)
- Enemy characteristics (e.g., tactics, training level)
- Weaponry

While it is not necessary for us to try to enumerate the different parameters that are associated with each of these, it is important to note that these factors are dynamic. Therefore, a system that predefines the parameter values by which these can change risks becoming obsolete as new threats, weapons, etc. are identified.

Solving this problem requires innovations to AI that previously have been difficult. Specifically, the underlying AI models that drive the simulation and the associated ITS and intelligent agent technologies need to be made transparent to and modifiable by the user. In this way, the user can literally re-invent his own training environment as the training requirement evolves. We see the key to this as being to make the underlying INKS that drives the technology modifiable by the user. We have begun to develop tools for accomplishing this on a separate project and envision transferring such tools to future work in this area.

References

Anderson, J.R. (1980). Cognitive psychology and its implications. San Francisco, CA: W.H. Freeman and Co.

Anderson, J.R., Farrell, R. and Sauers, R. (1984). Learning to program in LISP. Cognitive Science, 8, 87-129.

Bloom, B.S. (1984). The 2-sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. Educational Researcher, 13(6), 4-16.

Brna, P., Ohlsson, S., & Pain, H. (1993) (Eds.). Proceeding of Artificial Intelligent in Education '93. Charlottesville, VA: Association for the Advancement of Computing in Education.

Curtis, P. (1993). LambdaMOO Programmer's Manual, Xerox Palo Alto Research Center, August 1993.

Gentner, F.C., Cameron, J.A., & Crissey, M.J. (1997). Essential training needs and resources for simulation observer/controllers. Proceedings of the Interservice/Industry Training Systems and Education Conference. Arlington, VA: National Training Systems Association, 1997.

de Kleer, J. and Brown, J.S. (1981). Mental models of physical mechanisms and their acquisition. In J.R. Anderson (Ed.), Cognitive skills and their acquisition. Hillsdale, NJ: Erlbaum.

Dyer, R. (1996). Army Distance Learning Plan. Ft. Monroe, VA: Training Development and Analysis Directorate.

Greer, J. (1995) (Ed.) Proceeding of Artificial Intelligent in Education '95. Charlottesville, VA: Association for the Advancement of Computing in Education.

Leddo, J.M., Cardie, C.T., and Abelson, R.P. (1987). An integrated framework for knowledge representation and acquisition (Technical Report 87-14). Falls Church, VA: Decision Science Consortium, Inc., September 1987

Leddo, J., Cohen, M.S., O'Connor, M.F., Bresnick, T.A., and Marvin, F.F. (1990). Integrated knowledge elicitation and representation framework (Technical Report 90-3). Reston, VA: Decision Science Consortium, Inc.

Lesgold, A., Lajoie, S.P., Bunzo, M. and Eggan, G. (1990). A coached practice environment for an electronics troubleshooting job. In J. Larkin, R. Chabay and C. Cheftic (Eds.), Computer assisted instruction and intelligent tutoring systems: Establishing communication and collaboration. Hillsdale, NJ: Erlbaum.

Minsky, M. (1975). A framework for representing knowledge. In P. Winston (Ed.), The Psychology of computer vision. NY: McGraw-Hill.

NAEP 1992 Trends in Academic Progress (1992). Washington, DC: National Center for Education Statistics.

Newell, A. and Simon, H.A. (1972). Human problem solving. Englewood Cliffs, NJ: Prentice Hall.

Quillian, M.R. (1966). Semantic memory. Camridge, MA: Bolt, Beranek and Newman.

Schank, R.C. (1982). Dynamic memory: A theory of reminding and learning in computers and people. MA: Cambridge University Press.

Schank, R.C. & Abelson, R.P. (1977). Scripts, plans, goals and understanding. Hillsdale, NJ: Erlbaum.